



Module: Systèmes Embarqués

Chapitre 1: INTRODUCTION AUX SYSTÈMES EMBARQUÉS

El Kefi HLEL

Janvier 2014

DEFINITION D'UN SYSTÈME EMBARQUÉ

- Un système embarqué est défini comme un système **électronique** (matériel) et **informatique** (logiciel) autonome, spécialisé dans des tâches bien précises. Ses ressources sont généralement **limitées**.
- Système embarqué
 - Comprend une partie matérielle et une partie logicielle
 - Architecture spécifique dédiée. Effectuer des tâches précises
 - Certains doivent répondre à des contraintes de temps réel. D'autres ayant peu de contraintes au niveau performances permettent de simplifier le système et de réduire les coûts de fabrication
 - En contact avec l'environnement
 - N'est pas toujours un module indépendant. Le plus souvent il est intégré dans le dispositif qu'il contrôle
 - Possède des entrées/sorties spécifiques et réactives
- Le logiciel créé pour les systèmes embarqués est appelé **firmware**. Il est stocké dans une mémoire en lecture seule (ROM) ou dans une mémoire flash. Il fonctionne le plus souvent avec des ressources matérielles limitées : un petit, voire pas de clavier, un petit écran et peu de mémoire.

DOMAINES D'UTILISATION (1)



DOMAINES D'UTILISATION (2)

- Domaine grand public
 - smart phone, console de jeux, appareil photos, lecteur audio, décodeurs TV ...
- Moyens de transport
 - gestion moteur, ordinateur de bord, ABS, GPS, système navigation, système d'aide → automobiles, avions, trains, bateau, véhicule électrique, ...
- Equipement médicaux
 - imagerie (rayon X, IRM), endoscopie, monitoring, lasers, chirurgie, stimulateur cardiaque, ...
- Equipements de télécommunications
 - station mobile, routeur, gateway, satellite, ...
- Equipements industriels
 - commande, contrôle réparti, capteurs intelligents, ...
- ...

SPÉCIFICITÉS D'UN SYSTÈME EMBARQUÉ (1)

- Caractéristiques spécifiques:
 - Dédié pour une application spécifique
 - Coût réduit (de production)
 - Espace restreint (volume, capacité mémoire)
 - Capacité de calcul appropriée et adaptée
 - Exécution temps réel (la plupart des systèmes)
 - Fiabilité et sûreté de fonctionnement
 - Consommation d'énergie maîtrisée, voir très faible en cas d'utilisation des batteries
- Différences avec un ordinateur de bureau :
 - L'interface IHM (Interface Homme Machine) est adaptée selon l'application. Cela peut-être de simples leds et boutons jusqu'à un écran tactile. Généralement pas de clavier, et écran petit
 - Le système embarqué dispose de capteurs spécifiques pour son application: (température, courant, pression, module GSM-GPS, ...)

SPÉCIFICITÉS D'UN SYSTÈME EMBARQUÉ (2)

- Architecture d'un système embarqué :
 - Il est réalisé généralement autour d'un microcontrôleur
 - Un microcontrôleur est un système à processeur dans un seul chip avec de très bon rapport performance/prix
 - Comprend: processeur, mémoire, décodeur d'adresse, GPIO (entrées/sorties simples configurables) , contrôleur d'écran, contrôleur d'interruption, ...
 - Ensemble compact (volume optimisé)
 - Démarrage autonome du système (boot). Pas de disque dur, utilisation de mémoire flash
 - Généralement pas d'extension possible par l'ajout de cartes ou modules supplémentaires

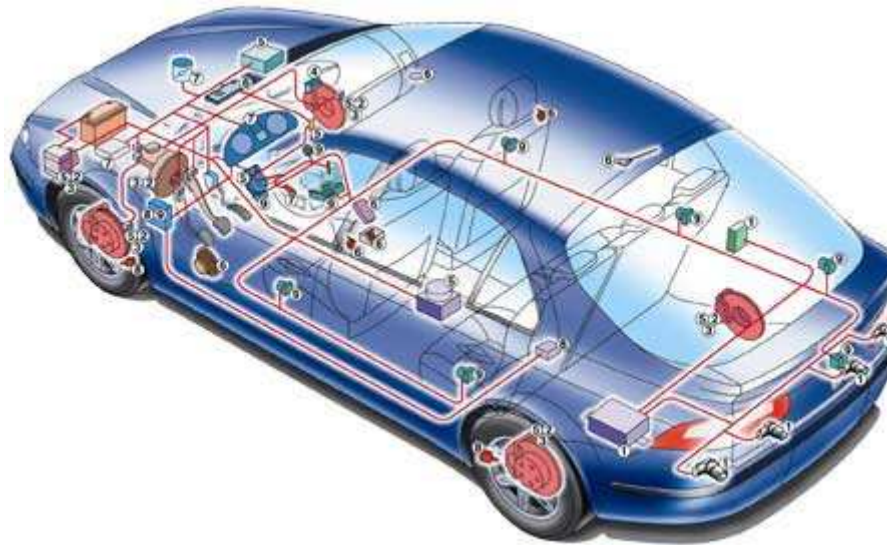
DÉVELOPPEMENT DE SYSTÈMES EMBARQUÉS

Le développement d'un système embarqué nécessite des connaissances à la fois en **électronique** et en **informatique**. Parmi le matériel nécessaire pour réaliser un système embarqué on trouve :

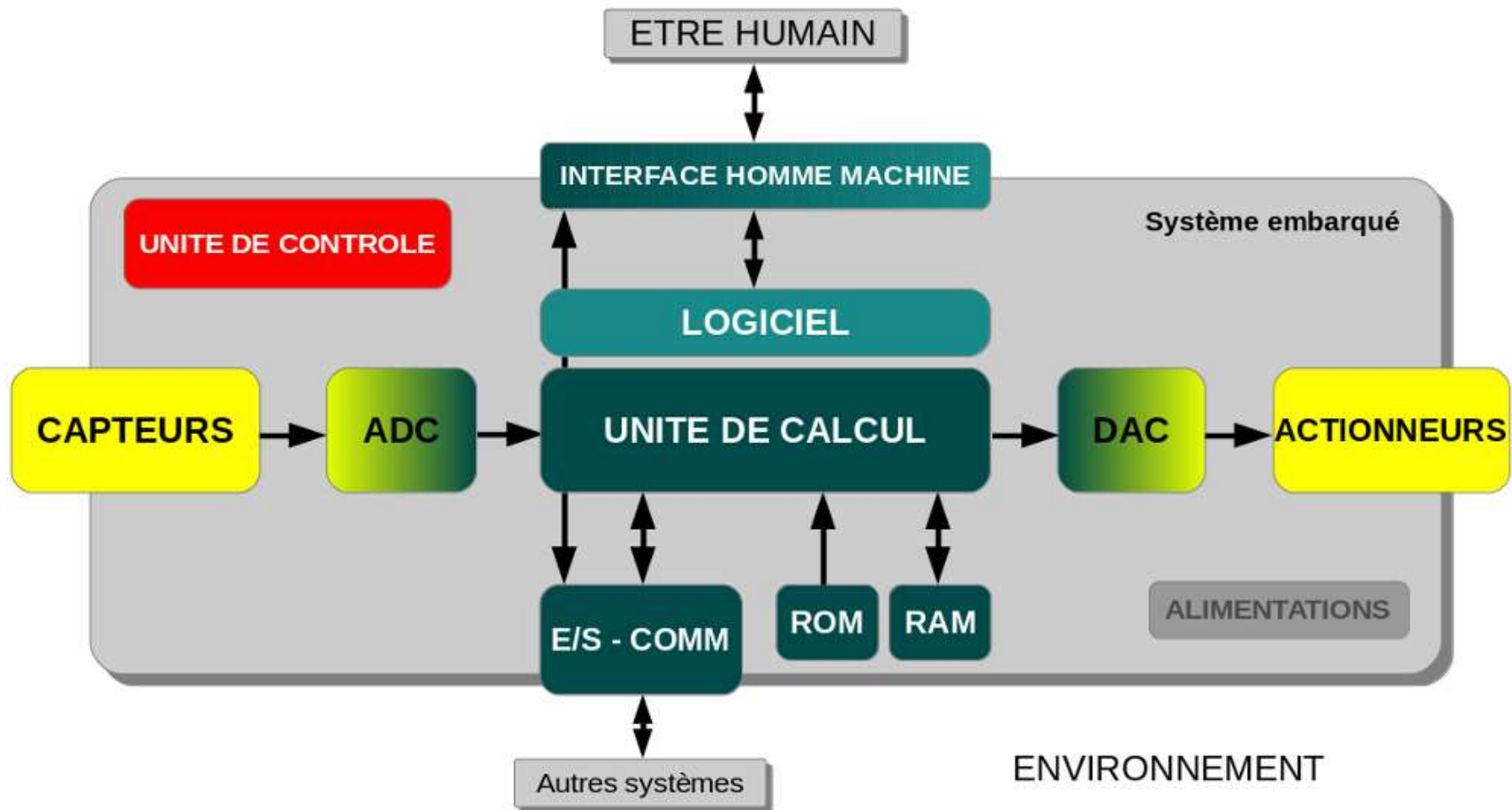
- La documentation sur les composants utilisés (datasheets)
- L'outillage de base de l'électronicien: fer à souder, ...
- Les outils d'analyse temporelle : oscilloscope, analyseur logique...
- Des composants de base: résistances, condensateurs, inductances,...
- Un microprocesseur ou un microcontrôleur
- Un compilateur croisé: pour compiler le noyau de l'OS, les applications, ...
- Un programmeur de microcontrôleur
- Un émulateur (*In Circuit Emulator*). Considéré comme l'équipement roi pour le debug matériel et logiciel
- Une sonde JTAG: pour la phase de développement

CARACTÉRISTIQUES

- Un système embarqué est principalement numérique pouvant intégrer une partie analogique (conditionnement de signaux, modulation, filtrage, ...)
- Système embarqué = une partie d'un système plus complexe
- Exemple: Dans automobile, entre 40 et 100 systèmes embarqués pour assurer :
 - La sécurité (ABS, EPS...)
 - Le confort (auto-radio, ouverture centralisée...)

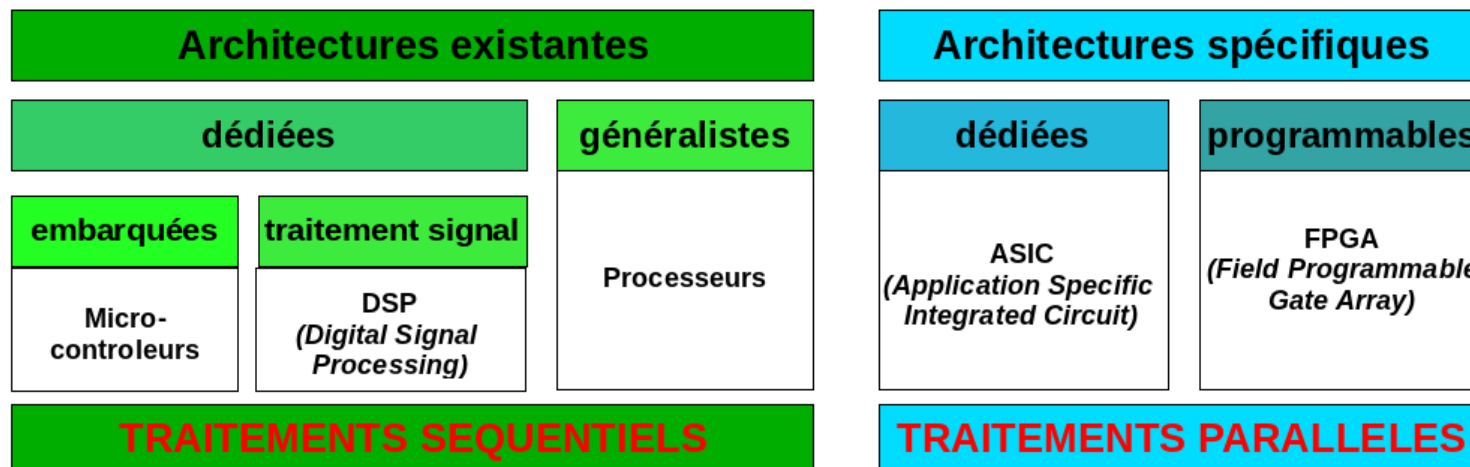


STRUCTURE D'UN SYSTÈME EMBARQUÉ



CONSTITUTION D'UN SYSTÈME EMBARQUÉ

- Pour concevoir un système embarqué "intelligent", deux solutions principales :
 - des architectures existantes (ex: microcontrôleurs)
 - des architectures spécifiques (FPGA, ASIC)



EVOLUTION DES PROCESSEURS

| Date | Description | Transistors |
|------|---|-----------------------------------|
| 1971 | Intel 4004, CPU 4 bits, 108 KHz Adressage 640 Bytes, 0.06 Mips | 2'300 10 μm |
| 1979 | Motorola 68000, CPU 32 bits, 8 MHz Adressage 16 MB, 0.7 Mips | 68'000 3.5 μm |
| 1990 | Motorola 68040, CPU 32 bits, 50MHz, Adressage 4 GB, MMU, FPU, 44Mips | 1.2 million 0.8 μm |
| 1999 | Pentium III, CPU 32 bits, 500MHz Adressage 4 GB, MMU, L1 cache 16KB | 9.5 million 0.25 μm |
| 2005 | Pentium D, CPU 64 bits, Dual cores, 2.8GHz | 290 million 0.09 μm |
| 2008 | Core i7, CPU 64 bits, 4 cores, 2.8GHz | 781 million |



EVOLUTION DES PROCESSEURS ARM

| Processor | Architecture | features |
|---------------|--------------|--|
| ARM7TDMI | ARMv4T | |
| ARM920 | ARMv4T | MMU |
| ARM926EJ-S | ARMv5E | MMU, DSP, Jazelle |
| ARM11 MPCore | ARMv6 | MPCore, MMU, DSP, Jazelle |
| Cortex-M1 | ARMv6-M | Intégration dans FPGA |
| Cortex-M3 | ARMv7-M | MPU (MMU), NVIC |
| Cortex-A8 | ARMv7-A | MMU, DSP, Jazelle, NEON + floating |
| ARM Cortex-A9 | ARMv7-A | MPCore, MMU, DSP, Jazelle, NEON + floating |

MMU: Memory Management Unit

MPCore: MultiProcessor Core

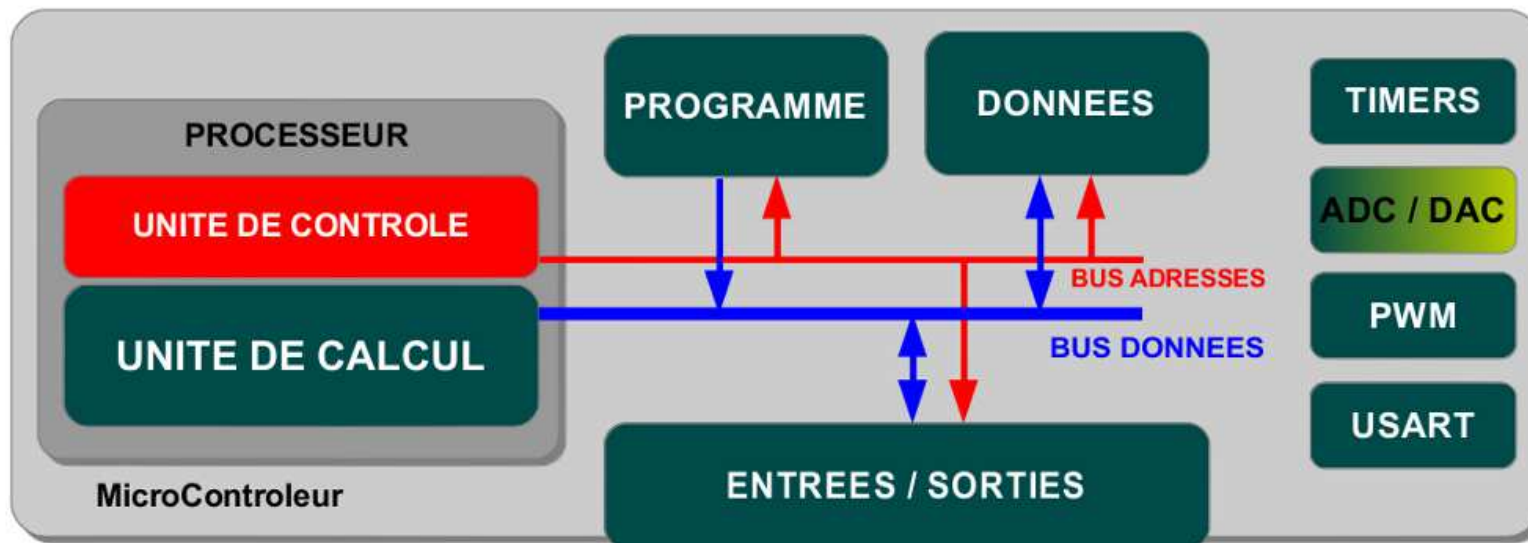
Jazelle: hardware Java Virtual Machine

NVIC: Nested Vectored Interrupt Controller

CONSTITUTION D'UN SYSTÈME EMBARQUÉ – MICROCONTROLEUR (1)

○ Microcontrôleur

- Unité de calcul séquentiel précablée
- Exécution d'une séquence d'instructions
- Entrées/Sorties spécifiques (numériques et analogiques)



CONSTITUTION D'UN SYSTÈME EMBARQUÉ – MICROCONTROLEUR (2)

○ Avantages

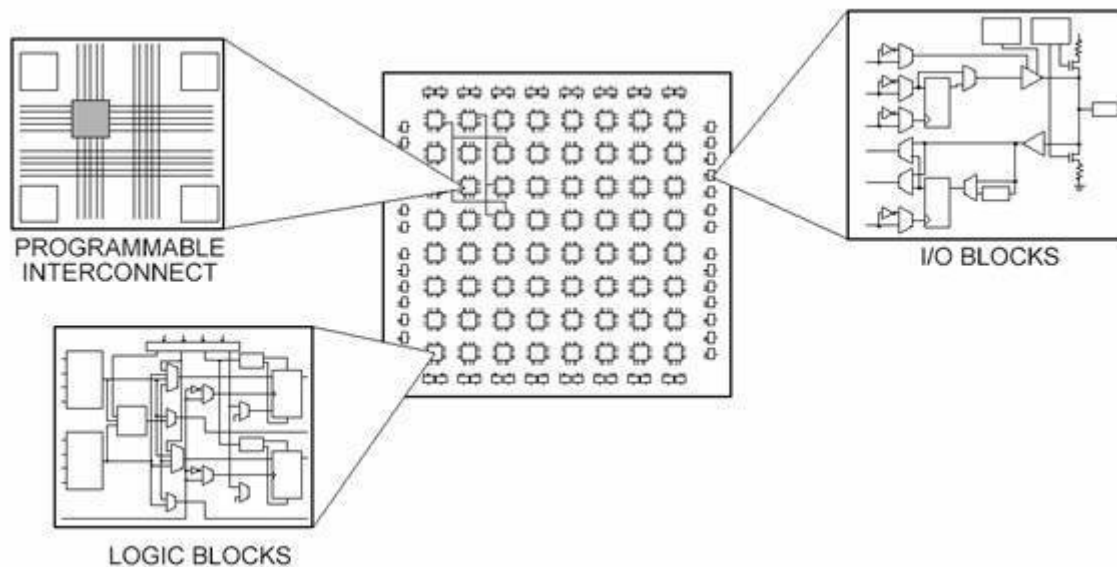
- Large choix de composants
- Modules déjà existants (ADC, PWM...)
- Gestion d'horloge – oscillateur interne
 - quelques MHz à quelques centaines de MHz
- Facilité de mise en œuvre

○ Inconvénients

- Exécution séquentielle des calculs. Moins rapide pour le traitement de données
- Instructions prédéfinies
- Utilisation réservée de certains modules (entrées-sorties limitées)

CONSTITUTION D'UN SYSTÈME EMBARQUÉ - FPGA ET ASIC (1)

- FPGA (Field Programmable Gate Array): électronique programmable
 - Interconnexion de composants logiques
 - Opérateurs logiques (logique combinatoire)
 - Bascules (logique séquentielle)
 - Description comportementale d'un système logique



- ASIC (Application Specific Integrated Circuit): un circuit intégré spécialisé. Il regroupe un grand nombre de fonctionnalités

CONSTITUTION D'UN SYSTÈME EMBARQUÉ - FPGA ET ASIC (2)

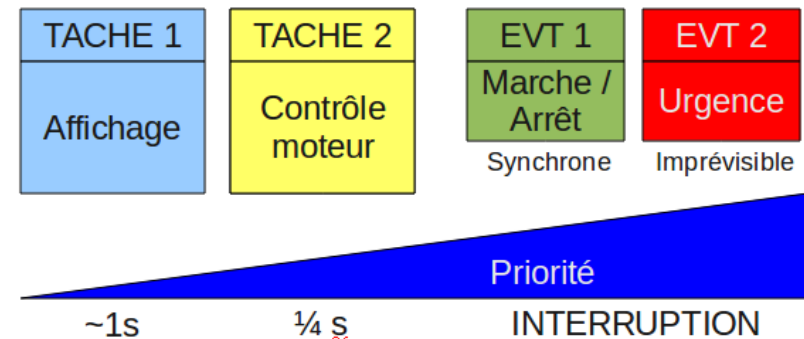
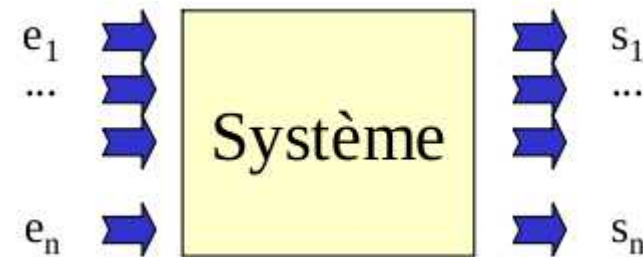
- Avantages
 - Exécution parallèle des calculs
 - Gestion d'horloge avancée
 - quelques MHz à quelques GHz
 - Reconfiguration dynamique
- Inconvénients
 - Aucun module précablé
 - Entrées-sorties numériques seulement

CHOIX DE L'ARCHITECTURE

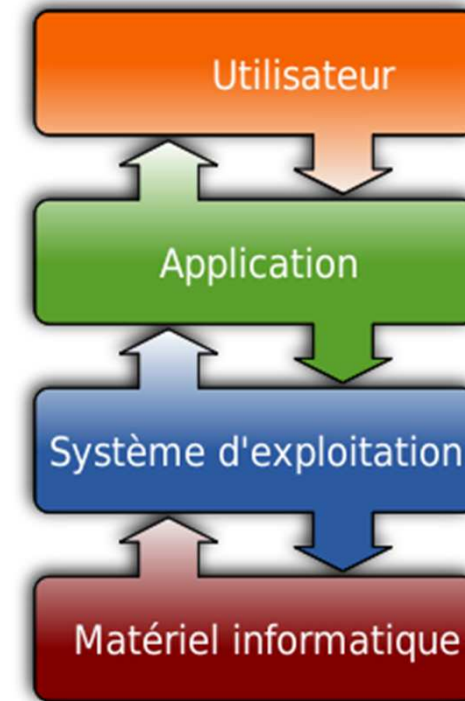
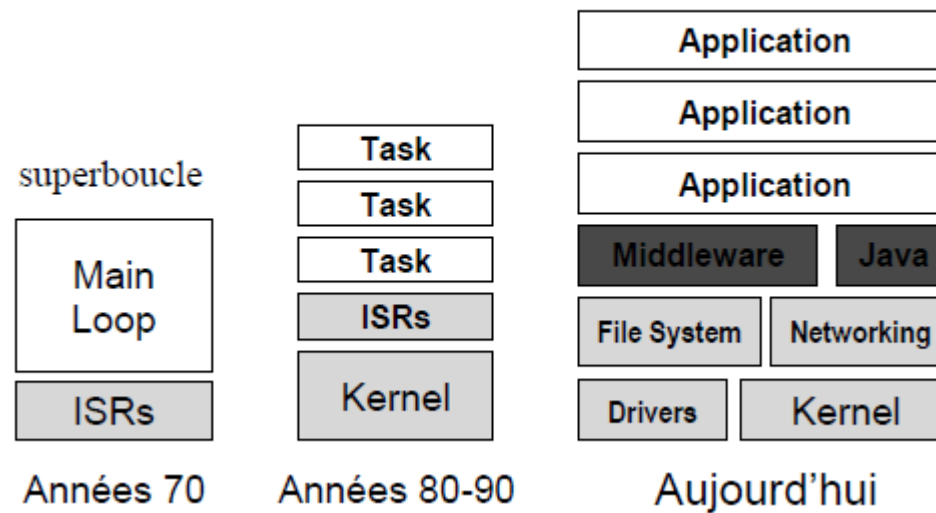
- Les critères à prendre en compte sont les suivants :
 - le nombre de fonctions à réaliser
 - le nombre d'entrées-sorties nécessaires
 - la vitesse d'exécution de ces fonctions
 - la consommation électrique
 - l'aspect temps réel et multitâches

NOTIONS IMPORTANTES

- Un système est caractérisé par :
 - sa relation entrees-sorties
 - son temps de réponse
- Gestion de tâches :
 - priorité
 - ordonnancement
- Gestion d'événements (indiquant l'evolution d'un système) :
 - synchrones
 - asynchrones (non predictibles) - interruptions
- SYSTEMES MULTITÂCHES / SYSTEMES TEMPS REEL



LES RTOS ET LES SYSTÈMES EMBARQUÉS



- Un système d'exploitation temps réel :
 - Apport du multitâche
 - Maîtrise des contraintes temporelles
 - Développement de pilotes de périphériques
 - Système de fichiers

PLAN DU COURS (1)

- **Chapitre 1** : Introduction aux systèmes embarqués
- **Chapitre 2** : Les processeurs
 - Architecture: Harvard / Von Neuman, Unités de calcul, Unité de contrôle, Séquenceur, Registres, ...
 - Composition & classification des processeurs
 - Les opérations du processeur, Parallélisme, Adressage
 - Les processeurs spécialisés (DSP, GPU,...)
- **Chapitre 3** : Les processeurs ARM
 - Présentation, Architecture & Technologies des processeurs ARM
 - Processeurs graphiques ARM
 - Divers processeurs ARM
 - Fabricants de processeurs ARM
- **Chapitre 4** : Les microcontrôleurs
 - Architecture interne
 - Les microcontrôleurs à base d'ARM (exemple STM32)
 - Environnements de programmation
 - Familles de microcontrôleurs

PLAN DU COURS (2)

○ **Chapitre 5 : Les FPGA & les ASICs**

- Programmation et mise en œuvre de l'électronique programmable
- Circuit logique programmable : Architecture matérielle, FPGA, CPLD, Applications, Conception du schéma logique, Procédés technologiques, Fabricants
- ASIC : un circuit intégré spécialisé, fil conducteur avec les circuits logiques programmables

○ **Chapitre 6 : Les DSPs**

- Définition, Instructions arithmétiques, Hardware Looping, Modes d'adressages spécialisés
- Caractéristiques & Programmation
- Fabricants

○ **Chapitre 7 : Les mémoires**

- Organisation des mémoires, Volatilité, Méthode d'accès
- Mémoires à accès séquentiel, Mémoires à accès aléatoire
- Classification des mémoires, ROM, RAM

○ **Chapitre 8 : Les périphériques et les interfaces**

- UART, I2C, SSP, CAN, USB, Ethernet

PLAN DU COURS (3)

- **Chapitre 9** : Systèmes d'exploitation pour les systèmes embarqués
 - Linux embarqué, Android, μ CLinux, les outils Microsoft pour l'embarqué
- **Chapitre 10** : Langages pour les embarqués et le temps réel
 - Assembleur, C, C++, Java

- **Mini-projet de réalisation pratique:**

Programmation sur un Microcontrôleur de STMicroelectronics.

Application d'utilisation de STM32. *(en parallèle avec l'avancement du cours après chapitre 4)*